Exploring the Steps of Verb Phrase Ellipsis

Zhengzhong Liu   Carnegie Mellon University

Edgar Gonzàlez   Google Inc.

Dan Gillick   Google Inc.

Verb Phrase Ellipsis: What is that?

1. When a verb constituent is partially or totally unexpressed.
2. But can be resolved through finding an antecedent verb constituent.
3. Verb Phrase Ellipsis (VPE) resolution
   a. An anaphoric process to recover the elided verb semantic.

Verb Phrase Ellipsis: What is that?

Factory payrolls **fell in September**.

So **did** the Federal Reserve Board's industrial-production index.

Source : fell in September

Target : did

Composer Marc Marder, a college friend of Mr. Lane's who earns his living playing the double bass in classical music ensembles, has prepared an exciting, eclectic score that **tells you what the characters are thinking and feeling far more precisely** than intertitles, or even words,**would**.

Source : tells you what the characters are thinking and feeling far more precisely

Target : would

Verb Phrase Ellipsis: Why are we doing it?

1. Verb Phrase Ellipsis (VPE) resolution fills up the missing local context of unspecified verbs.

2. Example: Dialogue Systems
   - ➢ Human: How can I get to Pittsburgh?
   - ➢ Computer: You could get there by plane.
   - ➢ Human: How do I **do** that?

   Verb phrase anaphoric analysis to understand the last question.

   - ➢ do -> get there by plane

Datasets

1. We use a dataset annotated on WSJ released by Bos and Spenader (2011)
2. We re-align the dataset annotated by Nielsen (2005) on the BNC corpus.

	Documents		VPE Instances	
	Train	Test	Train	Test
WSJ	1999	500	435	119
BNC	12	2	641	204

Basic Steps in Resolving VPE

1. Prior computational approaches describe the process as two steps *
2. Step 1: Target Ellipsis Detection
    a. Find out where a verb has been elided.
3. Step 2: Antecedent Selection
    a. Identify the antecedent phrase that can be used to recover the target elided verb.

* Nielsen (2005) describes an additional third step that rephrases the ellipsis word with the resolved verb phrase. This is out of the scope of this paper.

Antecedent Selection: What exactly is it?

1. The antecedent selection annotation is normally a verb phrase.
   a. The length of the phrase can be short or long.
2. To identify the phrase, we are doing two slightly different things:
   a. Find a verb that can recover the current ellipsis
   b. Find the correct constituent that cover the right amount of information
3. Example:
   a. Find the head "fell"
   b. Choose from possible spans
      i. "fell"
      ii. "fell in September"

Factory payrolls **fell in September**.

So **did** the Federal Reserve Board's industrial-production index.

Basic Steps: 3-step View

1. We consider splitting the process into 3 fine-grained steps:
    a. Target Detection.
    b. Antecedent Head Identification.
    c. Antecedent Boundary Detection.
2. Each step might have different characteristics.
    a. Hence different models may work better.
3. Making meaningful comparisons during learning
    a. Compare Head verb vs. Head verb
    b. Compare different trees rooted under the same head verb

Basic Steps: Target Detection

1. We consider only light verbs (be, do, have), modal verbs and "to" as candidates.
2. We use a logistic regression classifier to determine whether each candidate is a target.

Head	POS, Lemma, Dependency Label, Dependency Parent, Left and right words.
Dependent children	POS, Lemma, Dependency Label of these words.
3-word window	POS, Lemma, Dependency Label of these words.
Subject-verb inversion	Subject of the verb appears to its right.

Target Detection Performance

	WSJ			BNC		
	Prec	**Rec**	**F1**	**Prec**	**Rec**	**F1**
Oracle	100.00	93.28	96.52	100.00	92.65	96.18
Logistic	80.22	61.34	69.52	89.90	70.59	75.39
POS Base	42.62	43.70	43.15	35.47	35.29	35.38
Nielsen 2005	-	-	-	72.50	72.86	72.68

- ➢ Oracle is to use the gold standard on all candidates
- ➢ POS base is a POS baseline described in Nielsen (2005)
- ➢ Nielsen 2005 only reports their performance on BNC data.

Basic Steps: Antecedent Head Detection

1. To generate antecedent head candidates, we look at the following window:
   a. 3 immediately preceding sentences
   b. the same sentence of the candidate up to the target*
2. We then take all verbs (including modals and auxiliaries)
3. This generation step roughly follows Hardt (1992) and Nielsen (2005)

*In the Bos and Spenader (2011) corpus, there are 1% cataphoric cases, we ignore them in this work

Basic Steps: Antecedent Head Detection

1. We consider two different models
   a. A simple logistic classifier. (**Log**[H])
   b. A ranking based model. (**Rank**[H])
2. The ranking model is introduced since we consider that the features of different heads are comparable within each target, but might not be comparable cross target.
3. We adopt a ranking model with domination loss (Dekel et al., 2003)
   a. The ranking model allow us to specify preference over instances.
   b. Each correct antecedent is better than all of the incorrect candidates.

Basic Steps:
Antecedent Boundary Detection

1. Given an antecedent head, we then select from a set of potential antecedent boundaries.
2. These boundaries will result in partial or complete verb phrases.
3. We then learn to choose the best boundary with the same 2 models:
    a. A logistic regression classifier. (**Log^B**)
    b. A domination loss ranker. (**Rank^B**)

Antecedent Boundary Generation Algorithm

Data: a, the antecedent head

Data: t, the target

Result: B, the set of possible antecedent boundaries $(start, end)$

```
1  begin
2  │   a_s ⟵ SemanticHeadVerb(a);
3  │   E ⟵ {a_s} // the set of ending positions;
4  │   for ch ∈ RightChildren(a_s) do
5  │   │   e ← RightMostNode(ch);
6  │   │   if e < t ∧ ValidEnding(e) then
7  │   │   │   E ⟵ E ∪ {e}
8  │   B ⟵ ∅;
9  │   for e ∈ E do
10 │   │   B ⟵ B ∪ {(a, e)};
```

Example

In particular, Mr. Coxon says, businesses **are** paying out a smaller percentage of their profits and cash flow in the form of dividends than they **have** historically.

Given antecedent head word "are" and the target "have", the generated candidates are:

- are paying
- are paying out
- are paying out a smaller percentage of their profits and cash flow
- are paying out a smaller percentage of their profits and cash flow in the form of dividends

Features for Antecedents

1. Head detection
   a. Find parallel construction between the head and the target themselves.
   b. Find parallel construction of the context, especially the left-hand context.
2. Boundary detection
   a. Determine whether the phrase is well-formed.
   b. Find parallel construction of the right-hand context (since left hand side is determined)

Antecedent Features

Labels	The POS tag and dependency label of the antecedent head
	The POS tag and dependency label of the antecedent's last word
	The POS tag and lemma of the antecedent parent
	The POS tag, lemma and dependency label of within a 3 word around around the antecedent
	The pair of the POS tags of the antecedent head and the target, and of their auxiliary verbs
	The pair of the lemmas of the auxiliary verbs of the antecedent head and the target.
Distance	The distance in sentences between the antecedent and the target (clipped to 2)
	The number of verb phrases between the antecedent and the target (clipped to 5)
Match	Whether the lemmas of the heads, and words in the window (=2) before antecedent and target matches
	Whether the lemmas of the ith word before the antecedent and i−1th word before the target match respectively (for i ∈ {1, 2, 3}, with the 0th word of the target being the target itself)

Antecedent Features

Tree	Whether antecedent and target are dependent ancestor of each other.
	Whether antecedent and target share prepositions in their dependency tree.
	Whether the antecedent and the target form a comparative construction connecting by so, as or than.
	The dependency labels of the shared lemmas between the parse tree of the antecedent and the target.
	Label of the dependency between the antecedent and target (if exists).
	Whether the antecedent contains any descendant with the same lemma and dependency label as a descendant of the target.
Semantic	Whether the subjects of the antecedent and the target are coreferent Other.
Other	Whether the lemma of the head of the antecedent is be and that of the target is do (be-do match).
	Whether the antecedent is in quotes and the target is not, or vice versa.

Antecedent Head Detection Performance

	WSJ			BNC		
	Prec	**Rec**	**F1**	**Prec**	**Rec**	**F1**
Oracle	94.59	88.24	91.30	79.89	74.02	76.84
Rank	70.72	65.55	67.83	52.91	49.02	50.89
Previous Base*	67.57	63.03	65.22	39.68	36.76	38.17
Logistic	59.46	55.46	57.39	38.62	35.78	37.15

* Previous Base is the baseline that always use the immediate previous verb as antecedent head.

Antecedent Boundary Detection Performance (with oracle target and head)

	WSJ			BNC		
	Prec	**Rec**	**F1**	**Prec**	**Rec**	**F1**
Oracle	95.06	88.67	91.76	85.79	79.49	82.52
Logistic	89.47	83.46	86.36	81.10	75.13	78.00
Rank	83.96	78.32	81.04	75.68	70.12	72.79
Max Baseline	78.98	73.66	76.22	73.70	68.28	70.88

* Evaluation are done on token level Precision, Recall, F1

Joint Models

1. Jointly learn target + antecedent
   a. T+H
   b. T+H+B
   c. Since Logistic regression does not work well for H task, we modify our Ranker.
   d. Note that a Ranker will not give you a decision but only an order, so we add a NULL instance as decision boundary.,Correct target should be ranked higher than the NULL instance (this is previously used in coreference literatures).
2. Jointly learn the two steps of antecedent
   a. H+B
   b. This is simply using one model to predict both at the same time.

Joint Effect on Head Detection

	WSJ			BNC		
	Prec	Rec	F1	Prec	Rec	F1
Oracle	94.59	88.24	91.30	79.89	74.02	76.84
Rank	**70.72**	**65.55**	**67.83**	**52.91**	**49.02**	**50.89**
Previous Base	67.57	63.03	65.22	39.68	36.76	38.17
Logistic	59.46	55.46	57.39	38.62	35.78	37.15
Rank(H+B)	68.47	63.87	66.09	51.85	48.04	49.87
LOG(H+B)	39.64	36.97	38.26	30.16	27.94	29.01

Joint Effect on Boundary Detection

	WSJ			BNC		
	Prec	Rec	F1	Prec	Rec	F1
Ora(H)+Ora(B)	95.06	88.67	91.76	85.79	79.49	82.52
Rank(H)+Log(B)	64.11	59.80	61.88	47.04	43.58	45.24
Rank(H)+Rank(B)	63.90	59.60	61.67	49.11	45.50	47.24
Log(H)+Log(B)	53.49	49.89	51.63	34.77	32.21	33.44
Log(H)+Rank(B)	53.27	49.69	51.42	36.26	33.59	34.88
Rank(H+B)	**67.55**	**63.01**	**65.20**	**50.68**	**46.95**	**48.74**
LOG(H+B)	40.96	38.20	39.53	30.00	27.79	28.85

End-to-End Results

	WSJ			BNC		
	Prec	Rec	F1	Prec	Rec	F1
Ora(T)+Ora(H)+Ora(B)	95.06	88.67	91.76	85.79	79.49	82.52
Log(T)+Rank(H)+Rank(B)	52.68	40.28	45.65	**43.03**	**37.54**	**40.10**
Log(T)+Rank(H)+Log(B)	52.82	40.40	45.78	40.21	35.08	37.47
Log(T)+Log(H)+Rank(B)	49.45	37.82	42.86	33.12	28.90	30.86
Log(T)+Log(H)+Rank(B)	49.41	37.79	42.83	31.32	27.33	29.19
Pos(T)+Prev(H)+Max(B)	19.04	19.52	19.27	12.81	12.75	12.78
Log(T)+Rank(H+B)	**54.82**	**41.92**	**47.51**	41.86	36.52	39.01
Log(T)+Log(H+B)	38.85	29.71	33.67	26.11	22.78	24.33

Joint Model Does Not Work for Target+Antecedent?

1. Our ranking model that combines the joint target + antecedent selections are:
   a. Rank(T + H)
   b. Rank(T + H + B)
   c. Logistic(T + H)
   d. Logistic(T + H + B)
2. Both models fail to predict any target in WSJ corpus (a small number in BNC corpus)
   a. We found that the joint model exaggerates the class imbalance a lot
   b. The ranker need to consider all incorrect targets + all hypothesis antecedents

Future Directions

1. Shallow Semantic Information
   a. Most features used here are syntactic or shallow lexical features
   b. The coreference feature have a very small weight
   c. Further investigation must be performed to make use of shallow semantics.
2. Deeper Semantic Information
   a. Dan likes golf, and George does too. -> George likes golf.
   b. Dan likes his wife, and George does too. -> George likes his own wife.
   c. Actually the current token based evaluation scheme cannot tell the differences.

New Data Available

1. We've conducted experiments on a subset of data from the BNC corpus.
2. These data are annotated by Nielsen (2005)
3. We realign their annotation and converted them to Bos and Spenader format.
   a. This dataset contains more dialogue style annotations.
4. Available at http://github.com/hunterhector/VerbPhraseEllipsis

Summary

1. We propose a 3-step architecture and study the interaction of them.
2. We make systematic evaluation and trying to identify bottlenecks of the task.
3. We adapt an existing dataset and make it possible to use by the community.

Thank you and Q&A

References

Ofer Dekel, Christopher Manning, and Yoram Singer. 2004. Log-linear models for label ranking. *Advances in neural information processing systems*, 16:497–504.

Johan Bos and Jennifer Spenader. 2011. An annotated corpus for the analysis of VP ellipsis. *Language Resources and Evaluation*, 45(4):463–494.

Leif Arda Nielsen. 2005. *A corpus-based study of Verb Phrase Ellipsis Identification and Resolution*. Ph.D. thesis, King's College London.

Greg Durrett, David Hall, and Dan Klein. 2013. Decentralized Entity-Level Modeling for Coreference Resolution. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 114–124.

Altaf Rahman and Vincent Ng. 2009. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, number August, pages 968–977.

Daniel Hardt. 1998. Improving Ellipsis Resolution with Transformation-Based Learning. *AAAI Fall Symposium*:41–43.